REMARKS/ARGUMENT

Applicants have requested but have not yet received a certified copy of the application filed in the European Patent Convention on 11/29/2000 (00403344.5). Applicants will forward the certified copy to the Examiner as soon as they receive it.

Applicants have amended drawing Figure 1 by removing reference numerals 107, 108 and 109, as requested by the Examiner. Accordingly, Applicants have met the Examiner's requirement for new corrected drawings in compliance with 37 CFR 1.121(d).

Page 28, line 5 – page 29, line 13 has been deleted, as requested by the Examiner. Applicants are providing herewith a list of the same references on a form IDS, Form 1449.

Claims 9, 10, 13-21, 25 and 26 have been amended so as to not be directed to a "program per se". Accordingly, the 35 U.S.C. 101 rejection of Claims 9, 10, 13-21, 25 and 26 is overcome.

Claims 8, 20 and 21 have been amended so as to remove the trademark/trade name JAVA. Accordingly, the 35 U.S.C. 112, second paragraph, rejection is overcome.

Claims 1-27 stand rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent 6,571,388 B1 to Venkatraman et al., in view of US Patent 6,230,307 B1 to Davis et al. In light of the present amendments, Applicants respectfully traverse this rejection as set forth below.

AMENDMENTS TO THE DRAWINGS:

Responsive to the Examiner's requirement that corrected drawings in compliance with 37 CFR 1.121(d) be submitted, Applicants propose deleting reference numerals 107, 108 and 109 from Figure 1 since descriptions of the reference numerals is missing from the specification.

Copies of amended Figure 1 are included herewith. Applicants respectfully request approval.

Independent Claim 1, as amended, requires and positively recites, a method for **generating program source code** for translating high level code into instructions for a target processor~~s~~, the method comprising: "determining a program code characteristic corresponding to said target processor", "deriving one or more program code modules in accordance with said desired program code characteristic" and "**generating program source code for translating high level code into instructions for said target processor from said one or more program code modules**".

Independent Claim 9, as amended, requires and positively recites, a method for **creating program source code** for translating between high level code and instructions for a target processor, comprising the steps of: "determining a program code characteristic corresponding to said target processor", "selecting one or more predefined program code modules in accordance with said program code characteristic" and "**forming program source code for translating high level code into instructions for said target processor from said selected one or more predefined program code modules**".

Independent Claim 11, as amended, requires and positively recites, a data processing apparatus for **creating program source code** for translating between high level code and instructions for a target processor, the data processing apparatus being configured to: "determine a program code characteristic corresponding to said target processor identifier input to said data processing apparatus", "derive one or more program code modules in accordance with said program code characteristic" and "**create program source code for translating high level code into instructions for said target processor from said derived one or more program code modules**".

Independent Claim 13, as amended, requires and positively recites, an apparatus, comprising **at least one program source code module** of a plurality of **program source code modules** for translating between high level code and instructions for a target processor, **said at least one program source code module** corresponding to a characteristic of said target processor and being selected from said plurality of program code modules.

Independent Claim 22, as amended, requires and positively recites, a processor, configured in accordance with **program source code** comprising at least one **program source code module** of a plurality of **program source code modules**, for translating between high level code and instructions for a target processor, said at least one **program source code module** being in accordance with a characteristic of said target processor and selected from said plurality of **program source code modules**.

Independent Claim 23, as amended, requires and positively recites, a processor, configured by **program source code** comprising an **agglomeration of two or more program source code modules** of said plurality of said program code modules.

Independent Claim 24, as amended, requires and positively recites, a system comprising a first and second processor, said first and second processor configured in accordance with program code comprising **at least two program source code modules,** wherein the **first of said at least two program source code modules is arranged to translate high level code to instructions for said first processor** and **a second of said at least two program source code modules is arranged to translate high level code to instructions for said second processor**.

In proceedings before the Patent and Trademark Office, "the Examiner bears the burden of establishing a prima facie case of obviousness based upon the prior art". In re Fritch, 23 USPQ2d 1780, 1783 (Fed. Cir. 1992) (citing In re Piasecki, 745 F.2d 1468, 1471-72, 223 USPQ 785, 787-88 (Fed. Cir. 1984). "The Examiner can satisfy this burden **only by showing some objective teaching in the prior art or that knowledge generally**

available to one of ordinary skill in the art would lead that individual to combine the relevant teachings of the references", In re Fritch, 23 USPQ2d 1780, 1783 (Fed. Cir. 1992)(citing In re Fine, 837 F.2d 1071, 1074, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988)(citing In re Lalu, 747 F.2d 703, 705, 223 USPQ 1257, 1258 (Fed. Cir. 1988)).

Furthermore, "all words in a claim must be considered in judging the patentability of that claim against the prior art." In re Wilson, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970).

In contrast, the Venkatraman reference describes a system that pre-loads classes (30)(see also Abstract, lines 4-9 & 10-14) of one of several applications (32) with an embedded VM (34). Venkatraman selects a sub-set of application binary code (NOT source code) to embed with the virtual machine to avoid the cost of loading from disk, network or any other resources in the target device. While the present invention generates high level language to adapt source codes to each other, Venkatraman embeds a binary table to locate pre-loaded classes. As such, Venkatraman fails to teach or suggest, a method for **generating program source code** for translating high level code into instructions for a target processor, comprising: **"generating program source code for translating high level code into instructions for said target processor from said one or more program code modules"**, as required by Claim 1, OR a method for **creating program source code** for translating between high level code and instructions for a target processor, comprising: **"forming program source code for translating high level code into instructions for said target processor from said selected one or more predefined program code modules"**, as required by Claim 9, OR a data processing apparatus for **creating program source code** for translating between high level code and instructions for a target processor, the data processing apparatus being configured to: **"create program source code for translating high level code into instructions for said target processor from said derived one or more program code modules"**, as required by Claim 11.

Venkatraman further fails to teach or suggest an apparatus, comprising **at least one program source code module** of a plurality of **program source code modules** for translating between high level code and instructions for a target processor, **said at least one program source code module** corresponding to a characteristic of said target processor and being selected from said plurality of program code modules, as required by Claim 13, OR a processor, configured in accordance with **program source code** comprising at least one **program source code module** of a plurality of **program source code modules,** for translating between high level code and instructions for a target processor, said at least one **program source code module** being in accordance with a characteristic of said target processor and selected from said plurality of **program source code modules,** as required by Claim 22, OR a processor, configured by **program source code** comprising an **agglomeration of two or more program source code modules** of said plurality of said program code modules, as required by Claim 23, OR a system comprising a first and second processor, said first and second processor configured in accordance with program code comprising **at least two program source code modules,** wherein the **first of said at least two program source code modules is arranged to translate high level code to instructions for said first processor** and **a second of said at least two program source code modules is arranged to translate high level code to instructions for said second processor**, as required by Claim 25.

Whatever else the Davis reference discloses, it does not teach or suggest the above high-lighted elements in the claims. As such, Davis fails to provide any teaching or suggestion that would lead one having ordinary skill in the art to combine the teaching of Davis and Venkatraman in order to arrive at the teaching of the present invention.

Claims 2-8 stand allowable as depending (directly or indirectly) from allowable Claim 1 and including further limitations not taught or suggested by the references of record. Claim 10 stands allowable as depending from allowable Claim 9 and including further limitations not taught or suggested by the references of record. Claim 12 stands allowable as depending from allowable Claim 11 and including further limitations not taught or suggested by the references of record. Claims 14, 15 and 18-21 stand allowable as depending (directly or indirectly) from allowable Claim 13 and including further limitations not taught or suggested by the references of record.

Claim 2 further defines the method according to claim 1, by generating program source code for translating high level code into instructions **for one of a plurality of target processors**. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 1.

Claim 3 further defines the method according to claim 1, by further comprising forming agglomerated program source code from a plurality of program code modules in accordance with said desired program code characteristic. In addition to the reasons provided above for the allowance of Claim 1, there is no teaching whatsoever in Venkatraman that would lead one having ordinary skill in the art to "form agglomerated program source code from a plurality of program code modules in accordance with said desired program code characteristic. Similarly, there is no teaching in Davis that will overcome this deficiency in Venkatraman.

Claim 4 further defines the method according to claim 1, by further comprising deriving said program code modules in accordance with a desired functionality for said target processor. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 1.

Claim 5 further defines the method according to claim 2, wherein: "said step of determining comprises determining respective program code characteristics for respective ones of a plurality of target processors", "said step of deriving comprises deriving respective program code modules in accordance with said respective program code characteristics" and "said step of generating comprises generating program source code for translating high level code into instructions for said target processors from said program code modules". In addition to the reasons provided above for the allowance of Claim 2, Venkatraman discloses providing custom pre-loaded classes for application program for a single target device – not one of a plurality of target processors. Moreover, the Venkatraman and Davis references, alone or in combination, fail to teach or suggest multiple target processors.

Claim 6 further defines the method according to claim 1, wherein said step of deriving comprises selecting one or more pre-defined program code modules in accordance with said program code characteristic from a plurality of available program code modules. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 1.

Claim 7 further defines the method according to claim 1, wherein said program code provides a virtual machine for said target processor. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 1.

Claim 8 further defines the method according to claim 1, wherein said program code comprises elements of a programming language. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 1.

Claim 10 further defines the method according to claim 9, wherein said creating program source code for translating between high level code and instructions is for one of a plurality of target processors. In addition to the reasons provided above for the allowance of Claim 9, Venkatraman discloses providing custom pre-loaded classes for application program for a single target device – not one of a plurality of target processors. Moreover, the Venkatraman and Davis references, alone or in combination, fail to teach or suggest multiple target processors.

Claim 12 further defines the data processing apparatus according to claim 11, further configured for creating program source code for translating between high level code and instructions for one of a plurality of target processors. In addition to the reasons provided above for the allowance of Claim 11, Venkatraman discloses providing custom pre-loaded classes for application program for a single target device – not one of a plurality of target processors. Moreover, the Venkatraman and Davis references, alone or in combination, fail to teach or suggest multiple target processors.

Claim 14 further defines the apparatus of claim 13, by further comprising at least one additional program code modules for translating between high level code and instructions for respective ones of at least two target processors. In addition to the reasons provided above for the allowance of Claim 13, Venkatraman discloses providing custom pre-loaded classes for application program for a single target device – not one of a plurality of target processors. Moreover, the Venkatraman and Davis references, alone or in combination, fail to teach or suggest multiple target processors.

Claim 15 further defines the apparatus according to claim 14, wherein said at least two program code modules are selected from a plurality of predefined program code modules. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 14.

Claim 18 further defines the apparatus according to claim 13, wherein said program source code provides a virtual machine for said target processor. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 13.

Claim 19 further defines the apparatus according to claim 14, wherein said program source code provides a virtual machine for said target processor or processors. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 14.

Claim 20 further defines the apparatus according to claim 13, wherein said program source code comprises elements of a programming language. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 13.

Claim 21 further defines the apparatus according to claim 14, wherein said program source code comprises elements of a programming language. The Venkatraman and Davis references fail, alone or in combination, to teach or suggest this additional teaching in combination with the steps of Claim 1.

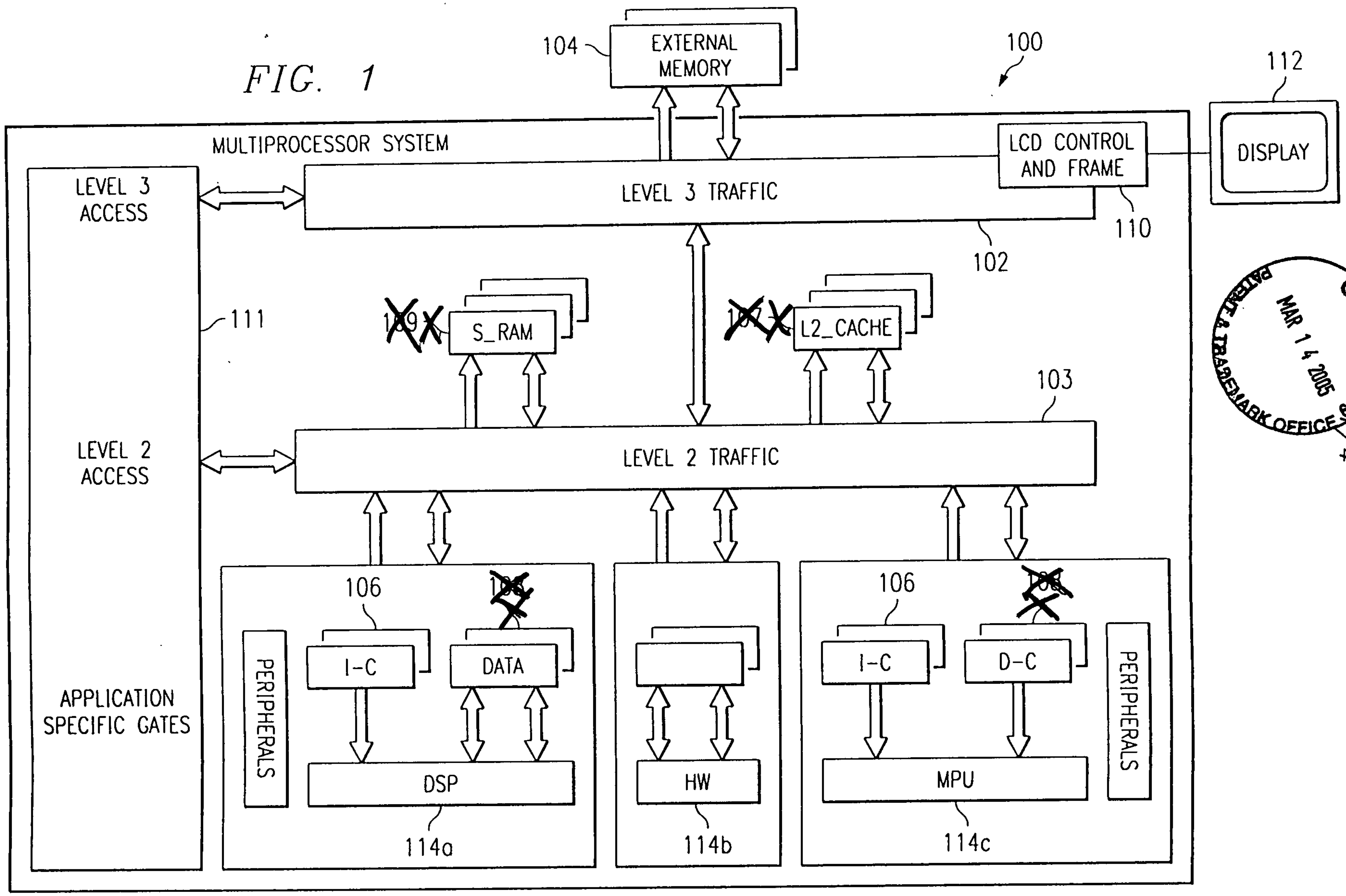
Claims 1-15 and 18-24 stand allowable over the references of record. Applicants respectfully request allowance of the application as the earliest possible date.

Respectfully submitted,

Ronald O. Neerings
Reg. No. 34,227
Attorney for Applicants

TEXAS INSTRUMENTS INCORPORATED
P.O. BOX 655474, M/S 3999
Dallas, Texas 75265
Phone: 972/917-5299
Fax: 972/917-4418

FIG. 1


100
104
EXTERNAL MEMORY
112
DISPLAY
MULTIPROCESSOR SYSTEM
LCD CONTROL AND FRAME
110
LEVEL 3 TRAFFIC
102
LEVEL 3 ACCESS
LEVEL 2 ACCESS
APPLICATION SPECIFIC GATES
111
~~109~~
S_RAM
~~107~~
L2_CACHE
103
LEVEL 2 TRAFFIC
106
~~108~~
I-C
DATA
PERIPHERALS
DSP
114a
HW
114b
106
~~108~~
I-C
D-C
PERIPHERALS
MPU
114c